



Europäisches Patentamt
European Patent Office
Office européen des brevets



0 405 926 A2

⑪ Publication number:

⑫

EUROPEAN PATENT APPLICATION

⑬ Application number: 90306996.1

⑬ Int. Cl. 5: G06F 11/16

⑭ Date of filing: 26.06.90

-S06F11/16

⑮ Priority: 30.06.89 US 374528

10 Woodward Road
Merrimack, New Hampshire 03054(US)

⑯ Date of publication of application:
02.01.91 Bulletin 91/01

Inventor: Goleman, William L.
6 Aspen Court
Nashua, New Hampshire 03062(US)
Inventor: Thiel, David W.
8 Ridgewood Drive
Amherst, New Hampshire 03031(US)

⑰ Designated Contracting States:
AT BE CH DE DK ES FR GB GR IT LI LU NL SE

⑯ Representative: Goodman, Christopher et al
Eric Potter & Clarkson St. Mary's Court St.
Mary's Gateate
Nottingham NG1 1LE(GB)

⑰ Applicant: DIGITAL EQUIPMENT
CORPORATION

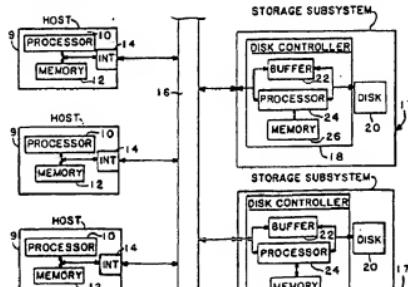
111 Powdermill Road
Maynard, MA 01754(US)

⑰ Inventor: Davis, Scott H.

⑲ Transferring data in a digital data processing system.

⑳ A system and method for transferring data from a first storage medium to a second storage medium, each of the storage media being divided into corresponding data blocks, the method comprising steps of: (a) reading data stored in a first data block in the first storage medium, the first data block initially constituting a current data block; (b) comparing data read in the current data block to data stored in a corresponding data block in the second storage

medium; (c) if the data compared in step b are identical, reading data stored in a different data block in the first storage medium, the different data block becoming the current data block, and returning to step b; (d) modifying the data stored in one of the storage media such that the data in the current data block is identical to the corresponding data in the second storage medium; and (e) rereading the data in the current data block and returning to step b.



TRANSFERRING DATA IN A DIGITAL DATA PROCESSING SYSTEM

BACKGROUND OF THE INVENTION

This invention relates to a device for transferring digital data between two storage devices in a digital data processing system. The preferred embodiment is described in connection with a system for establishing and maintaining one or more duplicate or "shadow" copies of stored data to thereby improve the availability of the stored data.

A typical digital computer system includes one or more mass storage subsystems for storing data (which may include program instructions) to be processed. In typical mass storage subsystems, the data is actually stored on disks. Disks are divided into a plurality of tracks, at selected radial distances from the center, and sectors, defining particular angular regions across each track, with each track and set of one or more sectors comprising a block, in which data is stored.

Since stored data may be unintentionally corrupted or destroyed, systems have been developed that create multiple copies of stored data, usually on separate storage devices, so that if the data on one of the copies is damaged, it can be recovered from one or more of the remaining copies. Such multiple copies are known as a "shadow set." In a shadow set, typically data that is stored in particular blocks on one member of the shadow set is the same as data stored in corresponding blocks on the other members of the shadow set. It is usually desirable to permit multiple host processors to simultaneously access (i.e., in parallel) the shadow set for read and write type requests ("I/O" requests).

A new storage device or "new member" is occasionally added to the shadow set. For example, it may be desirable to increase the number of shadow set members to improve the data availability or it may be necessary to replace a shadow set member that was damaged. Because all shadow set members contain the same data, when adding a new member, all of the data stored on the active members is copied to the new member.

Summary of the Invention

The invention generally features a system and method for transferring data from a first storage medium to a second storage medium and is used in the preferred embodiment to copy data from an active member of a shadow set to a new shadow set member. In the preferred embodiment, the first storage medium is available to one or more hosts

for I/O operations, and each of the storage media are divided into corresponding data blocks, the method generally including the steps of: (a) reading data stored in a first data block in the first storage medium, the first data block initially constituting a current data block, (b) comparing data read in the current data block to data stored in a corresponding data block in the second storage medium; (c) if the data compared in step b are identical, reading data stored in a different data block in the first storage medium, the different data block becoming the current data block, and returning to step b; (d) if the data compared in step b are not identical, transferring the data stored in the current data block to a corresponding data block in the second storage medium; and (e) rereading the data in the current data block and returning to step b.

In the preferred embodiment, the different data block is a data block adjacent to the current data block. Each data block in the first storage medium is compared to a corresponding data block in the second storage medium. Each of the storage media may be directly accessed by one or more host processors. The storage media may be disk storage devices.

The invention allows data to be copied from one storage media to a second storage media without interrupting I/O operations from one or more hosts to the shadow set. Therefore a shadowing system can be maintained that provides maximum availability to data with no interruption to routine I/O operations while providing consistent and correct results.

Other advantages and features of the invention will be apparent from the following detailed description of the invention and the appended claims.

Description of the Preferred EmbodimentsDrawings

We first briefly describe the drawings. Fig. 1 is a shadow set storage system according to the present invention.

Figs. 2-4 are data structures used with the invention.

Fig. 5 is a flow chart illustrating the method employed by the invention.

Structure and Operation

Referring to Fig. 1, a shadowing system utilizing the invention includes a plurality of hosts 9, each of which includes a processor 10, memory 12 (including buffer storage) and a communications interface 14. The hosts 9 are each directly connected through a communications medium 16 (e.g., by a virtual circuit) to two or more storage subsystems illustrated generally at 17 (two are shown).

Each storage subsystem includes a disk controller 18, that controls I/O requests to one or more disks 20, which form the members of the shadow set. Disk controller 18 includes a buffer 22, a processor 24 and memory 26 (e.g., volatile memory). Processor 24 receives I/O requests from hosts 9 and controls reads from and writes to disk 20. Buffer 22 temporarily stores data received in connection with a write command before the data is written to a disk 20. Buffer 22 also stores data read from a disk 20 before the data is transmitted to the host in response to a read command. Processor 24 stores various types of information in memory 12.

Each host 9 will store, in its memory 12, a table that includes information about the system that the hosts 9 need to perform many operations. For example, hosts 9 will perform read and write operations to storage subsystems 17 and must know which storage subsystems are available for use, what disks are stored in the subsystems, etc. As will be described in greater detail below, the hosts 9 will slightly alter the procedure for read and write operations if data is being transferred from one shadow set member to another. Therefore, the table will store status information regarding any ongoing operations. The table also contains other standard information.

While each storage subsystem may include multiple disks 20, the shadow set members are chosen to be disks in different storage subsystems. Therefore, hosts do not access two shadow set members through the same disk controller. This will avoid a "central point of failure." In other words, if the shadow set members have a common or central controller, and that controller malfunctions, the hosts will not be able to successfully perform any I/O operations. In the preferred system, however, the shadow set members are "distributed", and the failure of one device (e.g., one disk controller 18) will not inhibit I/O operations because they can be performed using another shadow set member accessed through another disk controller.

When a host wishes to write data to the shadow set, the host issues a command whose format is illustrated in Fig. 2A. The command includes a "command reference number" field that uniquely identifies the command, and a "unit number" field

write command for each disk that makes up the shadow set, using the proper unit number. The opcode field identifies that the operation is a write. The "byte count" field gives the total number of bytes contained in the data to be written and the "logical block number" identifies the starting location on the disk. The "buffer descriptor" identifies the location in host memory 12 that contains the data to be written.

10 The format of a read command is illustrated in Fig. 2B, and includes fields that are similar to the write command fields. For a read command, the buffer descriptor contains the location in host memory 12 to which the data read from the disk is to be stored.

15 Once a host transmits a read or write command, it is received by the disk controller 18 that serves the disk identified in the "unit number" field. For a write command, the disk controller will implement the write to its disk 20 and return an "end message" to the originating host, the format of the write command end message being illustrated in Fig. 3A. The end message includes a status field that informs the host whether or not the command was completed successfully. If the write failed the status field can include error information, depending on the nature of the failure. The "first bad block" field indicates the address of a first block on the disk that is damaged (if any).

20 For a read command, the disk controller will read the requested data from its disk and transmit the data to memory 12 of the originating host. An end message is also generated by the disk controller after a read command and sent to the originating host, the format of the read command end message being illustrated in Fig. 3B. The read command end message is similar to the end message for the write command.

25 As will be explained below, the system utilizes a "Compare Host" operation when transferring data between two shadow set members. The command message format for the Compare Host operation is shown in Fig. 4A. The Compare Host operation instructs the disk controller supporting the disk identified in the "unit number" field to compare the data stored in a section of host memory identified in the "buffer descriptor" field, to the data stored on the disk in the location identified by the "logical block number" and "byte count" fields.

30 The disk controller receiving the Compare Host command will execute the requested operation by reading the identified data from host memory, reading the data from the identified section of the disk, and comparing the data read from the host to the data read from the disk. The disk controller then issues an end message, the format of which is

end message will indicate whether the compared data was found to be identical.

When adding a new member to the shadow set (i.e., a new disk 20), the system chooses a host processor to carry out the processing necessary to provide the new member with all of the data stored in the active members of the shadow set. The host will choose one active member to function as a "source" and will sequentially copy all of the data from the source that differs from data in the new member, to the new member or "target." Using the method of invention, data is transferred to the new member without interrupting normal I/O operations between other hosts and the shadow set, while assuring that any changes to data in the shadow set made during the copy operation will propagate to the new member.

Specifically, the method of transferring data to a new member or target from a source involves sequentially reading data blocks, a "cluster" at a time (a cluster is a predetermined number of data blocks), from the source and comparing the read data to data stored at corresponding locations in the target. If the data is identical in the two corresponding clusters, then the next cluster is processed. Otherwise, the data read from the source is written to the target, and the host performs a similar comparison operation on the same cluster once again. The second comparison on the same cluster of data is necessary because the shadow set is available for I/O operations to other hosts during the process of adding a new member. In other words, while the new member is being added, I/O operations such as write operations from hosts in the system will continue to the shadow set. Read commands are performed to any active member, but not to the target since not all of the target's data is identical to the shadow set data. Since it is possible for a write operation to occur after the system reads data from the source and before it writes the data to the target, it is possible that obsolete data will be written into the target. I.e., if the data in the shadow set is updated just before the host writes data to the target, the data written to the target will be obsolete.

Therefore, the system will perform a second compare operation after it writes data to the target, to the same two corresponding data clusters. In this way, if the source has been updated, and the target was inadvertently written with obsolete data, the second comparison will detect the difference and the write operation will be repeated to provide the updated data to the target. Only after the two clusters are found to have identical data does the process move to the next data cluster.

It is theoretically possible for the same cluster to be compared and written many times. For example, if there was a particularly oft-written file or data

block that was being changed constantly, the source data cluster and target data cluster would always be inconsistent. To prevent the system from repeating the comparison and writing steps in an infinite loop, a "repeat cluster counter" is used to determine how many times the loop has been executed.

This counter is initialized to zero and is incremented after data is written from the source to the target. The counter is monitored to determine if it reaches a predetermined threshold number. The counter is reset when the system moves on to a new cluster. Therefore, if the same cluster is continually compared and written to the target, the counter will eventually reach the threshold value. The system will then reduce the size of the cluster and perform the comparison again. Reducing the size of the cluster will make it more likely that the clusters on the two members will be consistent since data in a smaller size cluster is less likely to be changed by a write from one of the hosts. When a successful comparison is eventually achieved, the cluster size is restored to its previous value.

As described above, I/O operations to the shadow set will continue while a new member is being added. However, hosts are required to perform write type operations in a manner that guarantees that while a new member is being added, all data written to logical blocks on the target disk will be identical to those contained on the source disk. If hosts issue I/O commands in parallel, as is normally done, it is possible that the data on the source and target will not be consistent after the copy method described above is implemented. To avoid possible data corruption, hosts shall ensure that write operations addressed to the source disk are issued and completed before the equivalent operation is issued to the target disk.

As explained above, each host stores a table that lists data that the host needs to operate properly in the system. For example, each table will include information regarding the disks that make up the shadow set, etc. The table also stores status information that informs the host whether or not a new member is being added to the shadow set. Therefore, before a host executes an I/O request to the shadow set it will check the status field in its table, and if the host determines that a new member is being added, the host will implement the special procedures discussed above for avoiding possible data corruption. The table is kept current by requiring hosts that begin the process of adding a new member, to send a message to every other host in the system, informing each host of the operation. A host that is controlling the addition of the new member will not begin the data transfer to the new member until it receives a confirmation from each host that each host has updated its table

to reflect the new status of the system. Similarly, a host controlling the addition of a new member will send a message to each host when the new member has been added, and has data that is consistent with the shadow set. Upon receiving this message, hosts will resume the normal practice of issuing I/O requests in parallel.

The method of the invention will now be explained in detail, with reference to the flow chart of Fig. 5. The host first initializes two counters: a "logical cluster counter" and the repeat cluster counter (step 1). The logical cluster counter is used to identify clusters in each shadow set member and, when initialized, will identify a first cluster of data. As the logical cluster counter is incremented, it sequentially identifies each cluster in the shadow set.

Next, the host selects one active shadow set member to serve as the source with the new member serving as the target (step 2). The host then issues a read command of the type illustrated in Fig. 2B to the disk controller serving the source (identified by the "unit number" field in the command) requesting that the data in the cluster identified by the logical cluster counter be read and transmitted to host memory (step 3).

The disk controller serving the source will receive the read command, will read the identified data from the source to its buffer 22 and will transmit the data to host memory 12, as well as issuing an end message (see Fig. 2B) informing the host that the read command was executed (step 4).

After the host receives the end message indicating that the data from the source is in host memory 12 (step 5), the host will issue a Compare Host command to the target to compare the data read from the source to the data stored in the same logical cluster in the target (step 6).

The target will receive the Compare Host command, and will perform the comparison, issuing an end message (see Fig. 4B) to the host indicating the result of the comparison (step 7).

The host receives the end message in step 8 and determines whether the data compared by the Compare Host command was identical. If the compared data was identical, then the logical cluster counter is tested to see if it is equal to a predetermined last number (indicating that all data clusters have been processed) (step 9). If the logical cluster counter is equal to the last number, then the process is finished. Otherwise, the logical cluster counter is incremented, the repeat cluster counter is reset (step 10), and the method returns to step 3 to begin processing the next cluster.

If the compared data was not identical (see

5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110 115 120 125 130 135 140 145 150 155 160 165 170 175 180 185 190 195 200 205 210 215 220 225 230 235 240 245 250 255 260 265 270 275 280 285 290 295 300 305 310 315 320 325 330 335 340 345 350 355 360 365 370 375 380 385 390 395 400 405 410 415 420 425 430 435 440 445 450 455 460 465 470 475 480 485 490 495 500 505 510 515 520 525 530 535 540 545 550 555 560 565 570 575 580 585 590 595 600 605 610 615 620 625 630 635 640 645 650 655 660 665 670 675 680 685 690 695 700 705 710 715 720 725 730 735 740 745 750 755 760 765 770 775 780 785 790 795 800 805 810 815 820 825 830 835 840 845 850 855 860 865 870 875 880 885 890 895 900 905 910 915 920 925 930 935 940 945 950 955 960 965 970 975 980 985 990 995 1000 1005 1010 1015 1020 1025 1030 1035 1040 1045 1050 1055 1060 1065 1070 1075 1080 1085 1090 1095 1100 1105 1110 1115 1120 1125 1130 1135 1140 1145 1150 1155 1160 1165 1170 1175 1180 1185 1190 1195 1200 1205 1210 1215 1220 1225 1230 1235 1240 1245 1250 1255 1260 1265 1270 1275 1280 1285 1290 1295 1300 1305 1310 1315 1320 1325 1330 1335 1340 1345 1350 1355 1360 1365 1370 1375 1380 1385 1390 1395 1400 1405 1410 1415 1420 1425 1430 1435 1440 1445 1450 1455 1460 1465 1470 1475 1480 1485 1490 1495 1500 1505 1510 1515 1520 1525 1530 1535 1540 1545 1550 1555 1560 1565 1570 1575 1580 1585 1590 1595 1600 1605 1610 1615 1620 1625 1630 1635 1640 1645 1650 1655 1660 1665 1670 1675 1680 1685 1690 1695 1700 1705 1710 1715 1720 1725 1730 1735 1740 1745 1750 1755 1760 1765 1770 1775 1780 1785 1790 1795 1800 1805 1810 1815 1820 1825 1830 1835 1840 1845 1850 1855 1860 1865 1870 1875 1880 1885 1890 1895 1900 1905 1910 1915 1920 1925 1930 1935 1940 1945 1950 1955 1960 1965 1970 1975 1980 1985 1990 1995 2000 2005 2010 2015 2020 2025 2030 2035 2040 2045 2050 2055 2060 2065 2070 2075 2080 2085 2090 2095 2100 2105 2110 2115 2120 2125 2130 2135 2140 2145 2150 2155 2160 2165 2170 2175 2180 2185 2190 2195 2200 2205 2210 2215 2220 2225 2230 2235 2240 2245 2250 2255 2260 2265 2270 2275 2280 2285 2290 2295 2300 2305 2310 2315 2320 2325 2330 2335 2340 2345 2350 2355 2360 2365 2370 2375 2380 2385 2390 2395 2400 2405 2410 2415 2420 2425 2430 2435 2440 2445 2450 2455 2460 2465 2470 2475 2480 2485 2490 2495 2500 2505 2510 2515 2520 2525 2530 2535 2540 2545 2550 2555 2560 2565 2570 2575 2580 2585 2590 2595 2600 2605 2610 2615 2620 2625 2630 2635 2640 2645 2650 2655 2660 2665 2670 2675 2680 2685 2690 2695 2700 2705 2710 2715 2720 2725 2730 2735 2740 2745 2750 2755 2760 2765 2770 2775 2780 2785 2790 2795 2800 2805 2810 2815 2820 2825 2830 2835 2840 2845 2850 2855 2860 2865 2870 2875 2880 2885 2890 2895 2900 2905 2910 2915 2920 2925 2930 2935 2940 2945 2950 2955 2960 2965 2970 2975 2980 2985 2990 2995 3000 3005 3010 3015 3020 3025 3030 3035 3040 3045 3050 3055 3060 3065 3070 3075 3080 3085 3090 3095 3100 3105 3110 3115 3120 3125 3130 3135 3140 3145 3150 3155 3160 3165 3170 3175 3180 3185 3190 3195 3200 3205 3210 3215 3220 3225 3230 3235 3240 3245 3250 3255 3260 3265 3270 3275 3280 3285 3290 3295 3300 3305 3310 3315 3320 3325 3330 3335 3340 3345 3350 3355 3360 3365 3370 3375 3380 3385 3390 3395 3400 3405 3410 3415 3420 3425 3430 3435 3440 3445 3450 3455 3460 3465 3470 3475 3480 3485 3490 3495 3500 3505 3510 3515 3520 3525 3530 3535 3540 3545 3550 3555 3560 3565 3570 3575 3580 3585 3590 3595 3600 3605 3610 3615 3620 3625 3630 3635 3640 3645 3650 3655 3660 3665 3670 3675 3680 3685 3690 3695 3700 3705 3710 3715 3720 3725 3730 3735 3740 3745 3750 3755 3760 3765 3770 3775 3780 3785 3790 3795 3800 3805 3810 3815 3820 3825 3830 3835 3840 3845 3850 3855 3860 3865 3870 3875 3880 3885 3890 3895 3900 3905 3910 3915 3920 3925 3930 3935 3940 3945 3950 3955 3960 3965 3970 3975 3980 3985 3990 3995 4000 4005 4010 4015 4020 4025 4030 4035 4040 4045 4050 4055 4060 4065 4070 4075 4080 4085 4090 4095 4100 4105 4110 4115 4120 4125 4130 4135 4140 4145 4150 4155 4160 4165 4170 4175 4180 4185 4190 4195 4200 4205 4210 4215 4220 4225 4230 4235 4240 4245 4250 4255 4260 4265 4270 4275 4280 4285 4290 4295 4300 4305 4310 4315 4320 4325 4330 4335 4340 4345 4350 4355 4360 4365 4370 4375 4380 4385 4390 4395 4400 4405 4410 4415 4420 4425 4430 4435 4440 4445 4450 4455 4460 4465 4470 4475 4480 4485 4490 4495 4500 4505 4510 4515 4520 4525 4530 4535 4540 4545 4550 4555 4560 4565 4570 4575 4580 4585 4590 4595 4600 4605 4610 4615 4620 4625 4630 4635 4640 4645 4650 4655 4660 4665 4670 4675 4680 4685 4690 4695 4700 4705 4710 4715 4720 4725 4730 4735 4740 4745 4750 4755 4760 4765 4770 4775 4780 4785 4790 4795 4800 4805 4810 4815 4820 4825 4830 4835 4840 4845 4850 4855 4860 4865 4870 4875 4880 4885 4890 4895 4900 4905 4910 4915 4920 4925 4930 4935 4940 4945 4950 4955 4960 4965 4970 4975 4980 4985 4990 4995 5000 5005 5010 5015 5020 5025 5030 5035 5040 5045 5050 5055 5060 5065 5070 5075 5080 5085 5090 5095 5100 5105 5110 5115 5120 5125 5130 5135 5140 5145 5150 5155 5160 5165 5170 5175 5180 5185 5190 5195 5200 5205 5210 5215 5220 5225 5230 5235 5240 5245 5250 5255 5260 5265 5270 5275 5280 5285 5290 5295 5300 5305 5310 5315 5320 5325 5330 5335 5340 5345 5350 5355 5360 5365 5370 5375 5380 5385 5390 5395 5400 5405 5410 5415 5420 5425 5430 5435 5440 5445 5450 5455 5460 5465 5470 5475 5480 5485 5490 5495 5500 5505 5510 5515 5520 5525 5530 5535 5540 5545 5550 5555 5560 5565 5570 5575 5580 5585 5590 5595 5600 5605 5610 5615 5620 5625 5630 5635 5640 5645 5650 5655 5660 5665 5670 5675 5680 5685 5690 5695 5700 5705 5710 5715 5720 5725 5730 5735 5740 5745 5750 5755 5760 5765 5770 5775 5780 5785 5790 5795 5800 5805 5810 5815 5820 5825 5830 5835 5840 5845 5850 5855 5860 5865 5870 5875 5880 5885 5890 5895 5900 5905 5910 5915 5920 5925 5930 5935 5940 5945 5950 5955 5960 5965 5970 5975 5980 5985 5990 5995 6000 6005 6010 6015 6020 6025 6030 6035 6040 6045 6050 6055 6060 6065 6070 6075 6080 6085 6090 6095 6100 6105 6110 6115 6120 6125 6130 6135 6140 6145 6150 6155 6160 6165 6170 6175 6180 6185 6190 6195 6200 6205 6210 6215 6220 6225 6230 6235 6240 6245 6250 6255 6260 6265 6270 6275 6280 6285 6290 6295 6300 6305 6310 6315 6320 6325 6330 6335 6340 6345 6350 6355 6360 6365 6370 6375 6380 6385 6390 6395 6400 6405 6410 6415 6420 6425 6430 6435 6440 6445 6450 6455 6460 6465 6470 6475 6480 6485 6490 6495 6500 6505 6510 6515 6520 6525 6530 6535 6540 6545 6550 6555 6560 6565 6570 6575 6580 6585 6590 6595 6600 6605 6610 6615 6620 6625 6630 6635 6640 6645 6650 6655 6660 6665 6670 6675 6680 6685 6690 6695 6700 6705 6710 6715 6720 6725 6730 6735 6740 6745 6750 6755 6760 6765 6770 6775 6780 6785 6790 6795 6800 6805 6810 6815 6820 6825 6830 6835 6840 6845 6850 6855 6860 6865 6870 6875 6880 6885 6890 6895 6900 6905 6910 6915 6920 6925 6930 6935 6940 6945 6950 6955 6960 6965 6970 6975 6980 6985 6990 6995 7000 7005 7010 7015 7020 7025 7030 7035 7040 7045 7050 7055 7060 7065 7070 7075 7080 7085 7090 7095 7100 7105 7110 7115 7120 7125 7130 7135 7140 7145 7150 7155 7160 7165 7170 7175 7180 7185 7190 7195 7200 7205 7210 7215 7220 7225 7230 7235 7240 7245 7250 7255 7260 7265 7270 7275 7280 7285 7290 7295 7300 7305 7310 7315 7320 7325 7330 7335 7340 7345 7350 7355 7360 7365 7370 7375 7380 7385 7390 7395 7400 7405 7410 7415 7420 7425 7430 7435 7440 7445 7450 7455 7460 7465 7470 7475 7480 7485 7490 7495 7500 7505 7510 7515 7520 7525 7530 7535 7540 7545 7550 7555 7560 7565 7570 7575 7580 7585 7590 7595 7600 7605 7610 7615 7620 7625 7630 7635 7640 7645 7650 7655 7660 7665 7670 7675 7680 7685 7690 7695 7700 7705 7710 7715 7720 7725 7730 7735 7740 7745 7750 7755 7760 7765 7770 7775 7780 7785 7790 7795 7800 7805 7810 7815 7820 7825 7830 7835 7840 7845 7850 7855 7860 7865 7870 7875 7880 7885 7890 7895 7900 7905 7910 7915 7920 7925 7930 7935 7940 7945 7950 7955 7960 7965 7970 7975 7980 7985 7990 7995 8000 8005 8010 8015 8020 8025 8030 8035 8040 8045 8050 8055 8060 8065 8070 8075 8080 8085 8090 8095 8100 8105 8110 8115 8120 8125 8130 8135 8140 8145 8150 8155 8160 8165 8170 8175 8180 8185 8190 8195 8200 8205 8210 8215 8220 8225 8230 8235 8240 8245 8250 8255 8260 8265 8270 8275 8280 8285 8290 8295 8300 8305 8310 8315 8320 8325 8330 8335 8340 8345 8350 8355 8360 8365 8370 8375 8380 8385 8390 8395 8400 8405 8410 8415 8420 8425 8430 8435 8440 8445 8450 8455 8460 8465 8470 8475 8480 8485 8490 8495 8500 8505 8510 8515 8520 8525 8530 8535 8540 8545 8550 8555 8560 8565 8570 8575 8580 8585 8590 8595 8600 8605 8610 8615 8620 8625 8630 8635 8640 8645 8650 8655 8660 8665 8670 8675 8680 8685 8690 8695 8700 8705 8710 8715 8720 8725 8730 8735 8740 8745 8750 8755 8760 8765 8770 8775 8780 8785 8790 8795 8800 8805 8810 8815 8820 8825 8830 8835 8840 8845 8850 8855 8860 8865 8870 8875 8880 8885 8890 8895 8900 8905 8910 8915 8920 8925 8930 8935 8940 8945 8950 8955 8960 8965 8970 8975 8980 8985 8990 8995 9000 9005 9010 9015 9020 9025 9030 9035 9040 9045 9050 9055 9060 9065 9070 9075 9080 9085 9090 9095 9100 9105 9110 9115 9120 9125 9130 9135 9140 9145 9150 9155 9160 9165 9170 9175 9180 9185 9190 9195 9200 9205 9210 9215 9220 9225 9230 9235 9240 9245 9250 9255 9260 9265 9270 9275 9280 9285 9290 9295 9300 9305 9310 9315 9320 9325 9330 9335 9340 9345 9350 9355 9360 9365 9370 9375 9380 9385 9390 9395 9400 9405 9410 9415 9420 9425 9430 9435 9440 9445 9450 9455 9460 9465 9470 9475 9480 9485 9490 9495 9500 9505 9510 9515 9520 9525 9530 9535 9540 9545 9550 9555 9560 9565 9570 9575 9580 9585 9590 9595 9600 9605 9610 9615 9620 9625 9630 9635 9640 9645 9650 9655 9660 9665 9670 9675 9680 9685 9690 9695 9700 9705 9710 9715 9720 9725 9730 9735 9740 9745 9750 9755 9760 9765 9770 9775 9780 9785 9790 9795 9800 9805 9810 9815 9820 9825 9830 9835 9840 9845 9850 9855 9860 9865 9870 9875 9880 9885 9890 9895 9900 9905 9910 9915 9920 9925 9930 9935 9940 9945 9950 9955 9960 9965 9970 9975 9980 9985 9990 9995 9999

ler, instructing the controller to write the data read from the source and sent to the host in step 4 to the section of the target identified by the logical cluster counter (i.e., the cluster in the target that corresponds to the cluster in the source from which the data was read) (step 11).

The disk controller serving the target receives the write command, reads the data from host memory and writes it to the section of the target identified by the current logical cluster counter and issues an end message to the host (step 12).

The host receives the end message indicating that the write has been completed (step 13), increments the repeat cluster counter and determines if the repeat cluster counter is greater than a threshold value (step 14). As explained above, if the same cluster is written a predetermined number of times, the repeat cluster counter will reach a certain value and the size of the cluster is reduced in the shadow set, the result of the first Compare Host operation will be negative and need not be performed. However, if the new disk being added was one that was in the shadow set in the recent past, then most of its data will be consistent with the data in the shadow set, and the Compare Host operation should be performed the first time.

Therefore, the invention shall be limited only by the scope of the appended claims.

50 Claims

1. A method of transferring data from a first storage medium to a second storage medium, said first storage medium being accessible to one or more host processors, each of said storage media being divided into corresponding data blocks, said method comprising the steps of:

said first storage medium, the first data block initially constituting a current data block;

- comparing the data read in said current data block to data stored in a corresponding data block in said second storage medium;
- (c) if the data compared in step b are identical, reading data stored in a different data block in said first storage medium, said different data block becoming the current data block, and returning to step b;
- (d) if the data compared in step b are not identical, modifying the data stored in one of said storage media such that the data in said current data block is identical to the corresponding data in said second storage medium; and
- (e) rereading the data in said current data block and returning to step b.

2. The method of claim 1 wherein said step of modifying comprises modifying the data in said second storage medium.

3. The method of claim 2 wherein said step of modifying comprises writing said data read from the current data block in said first storage medium to the corresponding data block in said second storage medium.

4. The method of claim 1 wherein said different data block is a data block adjacent to said current data block.

5. The method of claim 1 wherein each data block in said first storage medium is compared to a corresponding data block in said second storage medium.

6. The method of claim 1 wherein each of said storage media are members of a shadow set of storage media.

7. The method of claim 6 wherein each of said storage media may be directly accessed by a host processor.

8. The method of claim 6 wherein each of said storage media may be directly accessed by each of a plurality of host processors.

9. The method of claim 1 wherein said storage media are disk storage devices.

10. A method of managing a shadow set of storage media accessible by one or more host processors for I/O operations, comprising the steps of:

- carrying out successive comparisons of data stored in corresponding locations in a plurality of said storage media, respectively; and
- performing a management operation on at least one of said storage media, said management operation comprising, for each of said corresponding locations where said comparisons indicated that the data in said corresponding locations was not identical:
 - reading data from locations in one of said storage media and writing said data to corresponding locations in another of said storage media; and
 - comparing the data in said corresponding locations after said writing to determine if the data in said corresponding locations is identical.

11. An apparatus for managing a shadow set of storage media accessible by one or more host processors for I/O operations, comprising:

- means for carrying out successive comparisons of data stored in corresponding locations in a plurality of said storage media, respectively; and
- means for performing a management operation on at least one of said storage media, said management operation comprising, for each of said corresponding locations where said comparisons indicated that the data in said corresponding locations was not identical:
 - reading data from locations in one of said storage media and writing said data to corresponding locations in another of said storage media; and
 - comparing the data in said corresponding locations after said writing to determine if the data in said corresponding locations is identical.

12. A program for controlling one or more processors in a digital computer, the digital computer processing at least one process, which enables said processors to manage a shadow set of storage media accessible by one or more host processors for I/O operations, said program comprising:

- a comparison module for enabling one of said processors to carry out successive comparisons of data stored in corresponding locations in a plurality of said storage media, respectively; and a management module for enabling one of said processors to perform a management operation on at least one of said storage media, said management operation comprising, for each of said corresponding locations where said comparisons indicated that the data in said corresponding locations was not identical:
 - reading data from locations in one of said storage media and writing said data to corresponding locations in said other storage media; and
 - comparing the data in said corresponding locations after said writing to determine if the data in said corresponding locations is identical.

13. The method of claim 1 wherein each of said hosts will transmit any write requests to said storage media first to said first storage medium and, after said write request to said first storage medium has completed, will transmit said write request to said second storage medium.

14. The method of claim 1 wherein each of said host processors maintains a table including information relating to said data transfer.

15. The method of claim 10 wherein each of said storage media may be directly accessed by each

of a plurality of host processors.

16. The method of claim 10 wherein each of said hosts will transmit any write requests to said storage media first to said one of said storage media and, after said write request to said one of said storage media has completed, will transmit said write request to said another of said storage media. 5

17. The method of claim 10 wherein each of said host processors maintains a table including information relating to said management operation. 10

18. The method of claim 10 wherein step B(b) comprises rereading said data from said locations in said one of said storage media and comparing said reread data to data in said corresponding locations in said another of said storage media. 15

19. The method of claim 10 wherein steps B(a) and B(b) are repeated recursively for the same corresponding locations until the data stored in said corresponding locations is determined to be identical. 20

20. The apparatus of claim 11 wherein each of said hosts will transmit any write requests to said storage media first to said one of said storage media and, after said write request to said one of said storage media has completed, will transmit said write request to said another of said storage media. 25

21. The apparatus of claim 11 wherein each of said host processors maintains a table including information relating to said management operation. 30

22. The apparatus of claim 11 wherein each of said storage media may be directly accessed by each of a plurality of host processors.

23. The apparatus of claim 11 wherein said management operation comprises rereading said data from said locations in said one of said storage media after said writing and comparing said reread data to data in said corresponding locations in said another of said storage media. 35

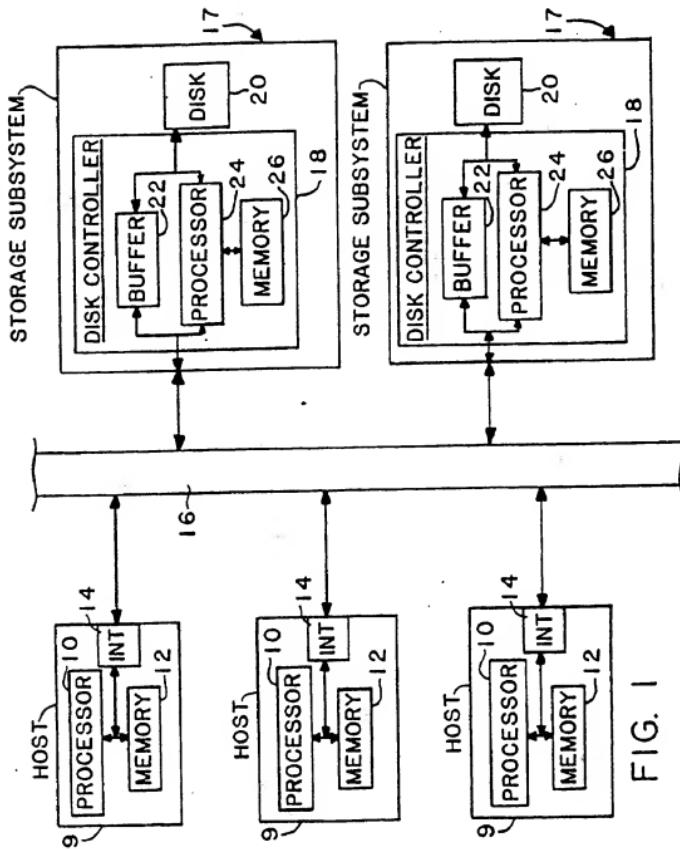
24. The apparatus of claim 11 wherein said means for performing said management operation repeats said reading and comparing recursively for the same corresponding locations until the data stored in said corresponding locations is determined to be identical. 40

45

50

55

eingereicht / Newly filed
Nouvellement déposé



—
E

Neu eingereicht / Newly filed
Nouvellement déposé

WRITE COMMAND MESSAGE FORMAT

COMMAND REFERENCE NUMBER		
RESERVED	UNIT NUMBER	
MODIFIERS	RESERVED	OPCODE
BYTE COUNT		
BUFFER		
DESCRIPTOR		
LOGICAL BLOCK NUMBER		

FIG. 2A

READ COMMAND MESSAGE FORMAT

COMMAND REFERENCE NUMBER		
RESERVED	UNIT NUMBER	
MODIFIERS	RESERVED	OPCODE
BYTE COUNT		
BUFFER		
DESCRIPTOR		
LOGICAL BLOCK NUMBER		

FIG. 2B

Neu eingereicht / Newly filed
Nouvellement déposé

WRITE END MESSAGE FORMAT

COMMAND REFERENCE NUMBER		
SEQUENCE NUMBER		UNIT NUMBER
STATUS	FLAGS	END CODE
BYTE COUNT		
----- UNDEFINED -----		
FIRST BAD BLOCK		

FIG. 3A

READ END MESSAGE FORMAT

COMMAND REFERENCE NUMBER		
SEQUENCE NUMBER		UNIT NUMBER
STATUS	FLAGS	END CODE
BYTE COUNT		
----- UNDEFINED -----		
FIRST BAD BLOCK		

FIG. 3B

Neu eingereicht / Newly filed
Nouvellement déposé

HOST COMPARE COMMAND FORMAT

COMMAND REFERENCE NUMBER		
RESERVED	UNIT NUMBER	
MODIFIERS	RESERVED	OPCODE
BYTE COUNT		
BUFFER		
DESCRIPTOR		
LOGICAL BLOCK NUMBER		

FIG. 4A

HOST COMPARE END MESSAGE FORMAT

COMMAND REFERENCE NUMBER		
SEQUENCE NUMBER		UNIT NUMBER
STATUS	FLAGS	END CODE
BYTE COUNT		
UNDEFINED		
FIRST BAD BLOCK		

FIG. 4B

*Neu eingereicht / Newly filed
Nouvellement déposé*

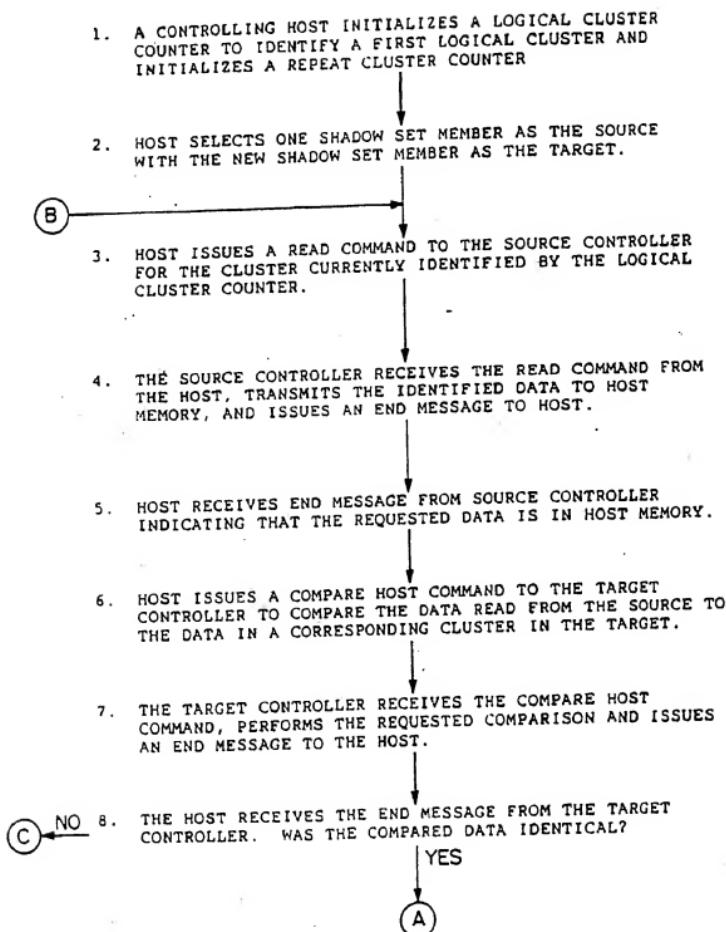


FIG. 5A

Neu eingereicht / Newly filed
Nouvellement déposé

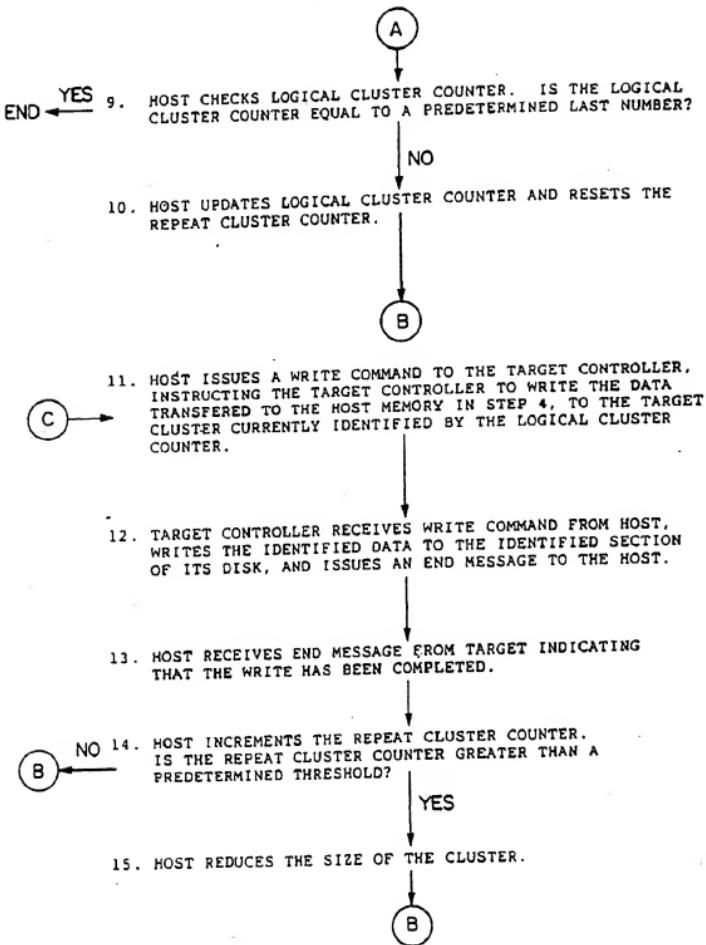


FIG. 5B



EUROPEAN PATENT APPLICATION

③ Application number: 90306996.1

⑤ Int. Cl. 5: G06F 11/14, G06F 11/20,
G06F 11/16

(zz) Date of filing: 26.06.90

Priority: 30-06-89 US 374528

⑩ Date of publication of application:

④ Designated Contracting States:

⑥ Date of deferred publication of the search report:
11.12.21. Bufl-Nr. 01/50

⑦ Applicant: **DIGITAL EQUIPMENT CORPORATION**
111 Powdermill Road
Maynard, MA 01754(US)

⑦ Inventor: Davis, Scott H.
10 Woodward Road
Merrimack, New Hampshire 03054(US)
Inventor: Goleman, William L.
6 Aspen Court
Nashua, New Hampshire 03062(US)
Inventor: Thiel, David W.
8 Ridgewood Drive
Amherst, New Hampshire 03031(US)

74 Representative: Goodman, Christopher et al
Eric Potter & Clarkson St. Mary's Court St.
Mary's Gate
Nottingham NG1 1LF (GB)

2) Transferring data in a digital data processing system.

⑦ A system and method for transferring data from a first storage medium to a second storage medium, each of the storage media being divided into corresponding data blocks, the method comprising steps of: (a) reading data stored in a first data block in the first storage medium, the first data block initially constituting a current data block; (b) comparing data read in the current data block to data stored in a corresponding data block in the second storage medium; (c) if the data compared in step b are identical, reading data stored in a different data block in the first storage medium, the different data block becoming the current data block, and returning to step b; (d) modifying the data stored in one of the storage media such that the data in the current data block is identical to the corresponding data in the second storage medium; and (e) rereading the data in the current data block and returning to step b.

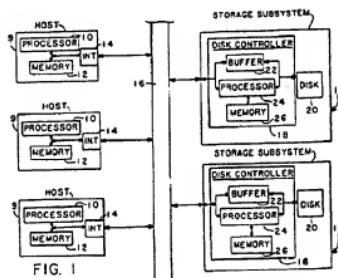


FIG. 1



European
Patent Office

EUROPEAN SEARCH
REPORT

Application Number

EP 90 30 6996

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (INT. CL.5)		
A	PATENT ABSTRACTS OF JAPAN vol. 9, no. 68 (P-344)(1791), 28 March 1985; & JP - A - 59201297 (NIPPON DENKI KK) 14.11.1984 "whole document" - - -	1	G 06 F 11/14 G 06 F 11/20 G 06 F 11/16		
A	US-A-4 686 620 (F.K. NG) "whole document" - - -	1			
TECHNICAL FIELDS SEARCHED (INT. CL.5)					
G 06 F					
The present search report has been drawn up for all claims					
Place of search	Date of completion of search	Examiner			
Berlin	17 September 91	ABRAM R			
CATEGORY OF CITED DOCUMENTS					
X: particularly relevant if taken alone Y: particularly relevant if combined with another document of the same category A: technological background D: non-written disclosure P: intermediate document T: theory or principle underlying the invention					
E: earlier patent document, but published on, or after the filing date D: document cited in the application L: document cited for other reasons A: member of the same patent family, corresponding document					